

WE CLAIM:

1. A method of updating a non-essential region stored in a memory device in a computer system, the method comprising:

building an image file, the image file comprising an essential region for storing program code required for booting the computer system and a non-essential region for storing optional program code for the computer system;

copying the image file to a memory device in the computer system; and

updating only the non-essential region in the memory device to update the non-essential data.

2. The method of claim 1, wherein the non-essential region in the image file comprises at least one non-essential block.

3. The method of claim 2, further comprising reserving at least one of a plurality of sectors in the memory device for storing the at least one non-essential block.

4. The method of claim 3, wherein updating the non-essential region in the memory device comprises mapping the at least one non-essential block to the at least one reserved sector in the memory device.

5. The method of claim 3, wherein updating the non-essential region in the memory device comprises mapping each non-essential block to a portion of the at least one reserved sector in the memory device.

6. The method of claim 5, wherein the portion of the at least one reserved sector in the memory device is a paragraph multiple.

7. The method of claim 1, wherein the program code in the essential region comprises a power-on self test (POST) routine.

8. The method of claim 2, wherein the at least one non-essential block comprises a header and at least one module.

9. The method of claim 8, wherein the header is located at the beginning of the at least one non-essential block.

10. The method of claim 8, wherein the header comprises a pointer to a first module in the at least one non-essential block.

11. The method of claim 8, wherein the at least one module comprises a module header and module data.

12. The method of claim 11, wherein the module header comprises a pointer to a next module in the at least one non-essential block.

13. The method of claim 11, wherein the module data comprises at least one of:

graphics data;
a language module; and
diagnostic tools.

14. A computer system for updating non-essential data in a memory device, comprising:

a memory device for storing an image file, the image file comprising an essential region for storing program code required for booting the computer system and a non-essential region for storing optional program code for the computer system;

a memory for storing a program containing code for updating the image file stored in a memory device; and

a processor, functionally coupled to the memory and associated with the memory device, wherein the processor is responsive to computer-executable instructions contained in the program and operative to:

copy the image file to the memory device; and

update only the non-essential region in the memory device to update the non-essential data in the memory device.

15. The computer system of claim 14, wherein the non-essential region in the image file comprises at least one non-essential block.

16. The computer system of claim 15, wherein the memory device comprises a plurality of sectors for storing the at least one non-essential block.

17. The computer system of claim 15, wherein the at least one non-essential block comprises a header and at least one module.

18. The computer system of claim 17, wherein the header is located at the beginning of the non-essential block.

19. The computer system of claim 17, wherein the at least one module comprises a module header and module data.

20. The computer system of claim 19, wherein the module data comprises program code.

21. The computer system of claim 19, wherein the module data comprises at least one of:

- graphics data;
- a language module; and
- diagnostic tools.

22. The computer system of claim 14, wherein the essential region in the image file comprises critical program code.

23. The computer system of claim 22, wherein the critical program code comprises a power-on self test (POST) routine.

24. A computer-readable medium having computer-executable instructions for performing steps comprising:

- building an image file, the image file comprising an essential region for storing program code required for booting the computer system and a non-essential region for storing optional program code for the computer system;

- copying the essential region and the non-essential region to the memory device; and

- updating only the non-essential region in the memory device.

25. The computer-readable medium of claim 24, wherein the non-essential region in the image file comprises at least one non-essential block.

26. The computer-readable medium of claim 25, further comprising reserving at least one of a plurality of sectors in the memory device for storing the at least non-essential block.

27. The computer-readable medium of claim 26, wherein updating only the non-essential region in the memory device comprises mapping the at least one non-essential block to the at least one reserved sector in the memory device.

28. The computer-readable medium of claim 26, wherein updating only the non-essential region in the memory device comprises mapping each non-essential block to a portion of the at least one reserved sector in the memory device.

29. The computer-readable medium of claim 28, wherein the portion of the at least one reserved sector in the memory device is a paragraph multiple.

30. The computer-readable medium of claim 24, wherein the program code in the essential region comprises a power-on self test (POST) routine.

31. The computer-readable medium of claim 25, wherein the at least one non-essential block comprises a header and at least one module.

32. The computer-readable medium of claim 31, wherein the header is located at the beginning of the at least one non-essential block.

33. The computer-readable medium of claim 31, wherein the header comprises a pointer to a first module in the at least one non-essential block.

34. The computer-readable medium of claim 31, wherein the at least one module comprises a module header and module data.

35. The computer-readable medium of claim 34, wherein the module header comprises a pointer to a next module in the at least one non-essential block.

36. The computer-readable medium of claim 34, wherein the module data comprises program code.

37. The computer-readable medium of claim 34, wherein the module data comprises at least one of:

- graphics data;
- a language module; and
- diagnostic tools.

38. A method of utilizing a non-essential region in a memory device for executing updated program code, the method comprising:

- searching the non-essential region in the memory device for at least one module, wherein the memory device comprises a plurality of modules containing program code for the computer system; and

- if the at least one module is found in the non-essential region, then executing the program code in the at least one module, wherein the at least one module in the non-essential region contains an updated version of the program code for the computer system.

39. The method of claim 38, further comprising:

- if the at least one module is not found in the non-essential region, then searching an essential region in the memory device for the at least one module, wherein the at least one module in the essential region contains a current version of the program code for the computer system ; and

- if the at least one module is found in the essential region, then executing the program code in the at least one module.

40. The method of claim 38, wherein the at least one module further comprises a module header, the module header comprising an identification of the program code contained in the at least one module.

41. The method of claim 38, wherein the updated version of the program code contained in the non-essential region comprises updated program code for a BIOS in the computer system.